## 10 IONet - I/O & Network Programing Aneka Soal Ujian Sistem Operasi Rahmat M. Samik-Ibrahim et.al.

© 2016 - 2018 — Rev: 14 - 07-Mar-2018. Silakan mengubah, memperbanyak, serta mendistribusikan dokumen ini selama tidak menghapus ketentuan ini. URL: http://rms46.vlsm.org/2/205.pdf

## 1. **2016-1**

Lingkari atau beri silang huruf "B" jika betul, dan "S" jika salah.

```
001 /* (c) 2015-2016 Rahmat M. Samik-Ibrahim
002 * R: 05-Jun-2016 -- This is free software */
003
004 #include <stdio.h>
005 #include <string.h>
006 #include <unistd.h>
007 #include <fcntl.h>
008 #include <sys/types.h>
009 #include <sys/stat.h>
010
011 char *string = "ABCD\n";
012 void main() {
013
       int
             fileDescriptor;
       close(STDOUT_FILENO);
014
015
       fileDescriptor = open ("output.txt", O_RDWR|O_CREAT|O_TRUNC, 0644);
                       "%s", string);
016
       printf (
017
       write(fileDescriptor, string, strlen(string));
018 }
```

- **B** / **S** Tanpa baris 004 009, program akan tetap dapat dikompilasi tanpa kesalahan (error).
- ${f B}$  /  ${f S}$  Pointer "string" (baris 011) merupakan variabel global.
- **B / S** Deklarasi "void main()" (baris 12) artinya: tidak ada "passing argument" ke dalam fungsi main().
- **B / S** Pada saat program dieksekusi, secara otomatis file descriptor dari streams stdin=0 (STDIN\_FILENO), stdout=1 (STDOUT\_FILENO), dan stderr=2 (STDERR\_FILENO).
- **B** / **S** Baris 14 akan menutup stream STDOUT\_FILENO (1).
- ${f B}$  /  ${f S}$  Nilai "file Descriptor" = 1 (baris 15), akibat baris no 14.
- ${\bf B}$  /  ${\bf S}$  Jika berkas "output.txt" tidak ada (baris 15), maka fungsi open() akan membuat berkas "output.txt" baru.
- **B** / **S** Jika sudah ada berkas "output.txt" (baris 15), maka fungsi open() akan membuka berkas dengan mode menambah (append).
- **B** / **S** Fungsi "printf()" (baris 16) akan menulis "ABCD\n" ke layar monitor.

Isi semula berkas "output.txt" ialah "XXXX\n"; maka setelah program dieksekusi akan berisi:

## 2. **2017-1**

C	Programing								
001 /*	020 static char* str1 = "AABB\n";								
002 * (c) 2017 Rahmat M. Samik-Ibrahim	021 static char* str2 = "CCDD\n";								
This is free software	022 static char* str3 = "EEFF\n";								
003 * REV00 Thu Mar 30 16:56:54 WIB 2017	023								
004 * START Thu Mar 30 16:56:54 WIB 2017	024 void main(void) {								
005 *	025 int fd1, fd2, fd3;								
006 * fd2=dup(fd1) duplicates fd1 to fd2	026 /* STDIN=0, STDOUT=1, STDERR=2, therefore								
007 * O_RDWR Open the file so that it can be read	027 fd1, fd2, fd3 will be 3, 4, and 5 $*/$								
from and written to.	028 fd1 = open (FILE, 0_TRUNC   0_RDWR   0_CREAT, 0644);								
008 * O_TRUNC Initially clear all data from the file.	029 fd2 = open (FILE, O_TRUNC   O_RDWR   O_CREAT, 0644);								
009 * O_CREAT If the file does not exist, create it.	fd3 = dup(fd2);								
010 */	031 printf("fd1=%d, fd2=%d, fd3=%d\n", fd1, fd2, fd3);								
011	032 write(fd1, str1, strlen(str1));								
012 #include <stdio.h></stdio.h>	033 write(fd2, str2, strlen(str2));								
013 #include <unistd.h></unistd.h>	<pre>034 write(fd3, str3, strlen(str3));</pre>								
014 #include <sys types.h=""></sys>	035 close(fd1);								
015 #include <sys stat.h=""></sys>	036 close(fd2);								
016 #include <fcntl.h></fcntl.h>	037 close(fd3);								
017 #include <string.h></string.h>	038 }								
018 #define FILE "uts2017-1.txt"									
Program Output (Line 031):									
Content of file "uts2017-1.txt"									

## 3. **2017-2**

	C Programing I/O								
001 /*	015 #include <sys stat.h=""></sys>								
002 * (c) 2017 Rahmat M. Samik-Ibrahim	016 #include <fcntl.h></fcntl.h>								
003 * http://rahmatm.samik-ibrahim.vlsm.org/	017 #include <string.h></string.h>								
004 * This is free software.	018 #define FILE "file.txt"								
005 * REV00 Wed Oct 18 18:20:27 WIB 2017	019								
006 * START Wed Oct 18 18:20:27 WIB 2017	020 void main(void) {								
007	021 int fd1, fd2;								
008 write (fd, "string", string_lenght);	022 fd1 = open (FILE, O_RDWR   O_CREAT   O_TRUNC, 0644);								
009 duplicate = dup (fd);	023 fd2 = dup(fd1);								
010 */	024 write (fd1, "012345\n", 5);								
011	025 write (fd2, "abcdef\n", 5);								
012 #include <stdio.h></stdio.h>	026 close(fd1);								
013 #include <unistd.h></unistd.h>	027 close(fd2);								
014 #include <sys types.h=""></sys>	028 }								
Inside FILE (file.txt)									